

ADA064324

DDC FILE COPY

ARI TECHNICAL REPORT  
TR-78-A40

**Modeling of Tactical Events by Interactive  
Graphics: Approach, Interface Design,  
and System Design**

**LEVEL**

by  
G. W. Irving, Lee M. Farrell, and Gary Lindquist  
Integrated Sciences Corporation  
Santa Monica, California 90401

**December 1978**

**Contract DAHC 19-77-C-0022**

Monitored technically by S. M. Halpin and F. L. Moses  
Human Factors Technical Area, ARI

**Prepared for**

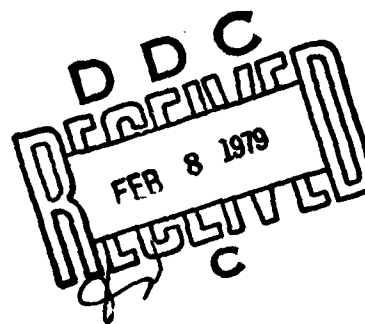


**U.S. ARMY RESEARCH INSTITUTE  
for the BEHAVIORAL and SOCIAL SCIENCES  
5001 Eisenhower Avenue  
Alexandria, Virginia 22333**

ORIGINAL CONTAINS COLOR PLATES: ALL DDC  
REPRODUCTIONS WILL BE IN BLACK AND WHITE.

Approved for public release; distribution unlimited

79 02 05 036



# **U. S. ARMY RESEARCH INSTITUTE FOR THE BEHAVIORAL AND SOCIAL SCIENCES**

**A Field Operating Agency under the Jurisdiction of the  
Deputy Chief of Staff for Personnel**

**JOSEPH ZEIDNER**  
Technical Director

**WILLIAM L. HAUSER**  
Colonel, US Army  
Commander

---

Research accomplished for  
the Department of the Army

Integrated Sciences Corporation

## **NOTICES**

**DISTRIBUTION:** Primary distribution of this report has been made by ARI. Please address correspondence concerning distribution of reports to: U. S. Army Research Institute for the Behavioral and Social Sciences, ATTN: PERI-P, 5001 Eisenhower Avenue, Alexandria, Virginia 22333.

**FINAL DISPOSITION:** This report may be destroyed when it is no longer needed. Please do not return it to the U. S. Army Research Institute for the Behavioral and Social Sciences.

**NOTE:** The findings in this report are not to be construed as an official Department of the Army position, unless so designated by other authorized documents.

ARI REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM	
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER	
19 TR-78-A40	2 Final rept. 14 apr 77-13 Feb 78		
4. TITLE (and Subtitle)	5. TYPE OF REPORT & PERIOD COVERED		
6 Modeling of Tactical Events by Interactive Graphics: Approach, Interface Design, and System Design.	Final 4/14/77 to 2/13/78		
7. AUTHOR	8. PERFORMING ORG. REPORT NUMBER		
10 G. W. Irving, Lee M. Farrell, and Gary Lindquist	14 ISC-255-1		
9. PERFORMING ORGANIZATION NAME AND ADDRESS	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS		
Integrated Sciences Corporation 1640 Fifth Street Santa Monica, CA 90401	16 2Q162722A765		
11. CONTROLLING OFFICE NAME AND ADDRESS	12. REPORT DATE		
U.S. Army Research Institute for the Behavioral and Social Sciences, 5001 Eisenhower Avenue Alexandria, VA 22333	11 December 1978		
13. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	14. NUMBER OF PAGES		
12 66p.	59		
15. SECURITY CLASS. (of this report)		16. SECURITY CLASS. (of this report)	
Unclassified		Unclassified	
17. DISTRIBUTION STATEMENT (of this Report)			
Approved for public release; distribution unlimited.			
18. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)			
--			
19. SUPPLEMENTARY NOTES			
The research was monitored technically by Stanley M. Halpin and Franklin L. Moses of the Human Factors Technical Area, ARI.			
20. KEY WORDS (Continue on reverse side if necessary and identify by block number)			
Decision Aids Army Tactics Intelligence Data Analysis Interactive Graphics		Intelligence Collection Planning Command and Control	
21. ABSTRACT (Continue on reverse side if necessary and identify by block number)			
The potential role for computer graphics in interactive tactical modeling was investigated using a four-color vector graphics display. The display enabled a user to define terrain features, order of battle, and the movements and actions of units. Interactive devices included a light pen, track ball, and function keyboard. The model emphasized tactical planning, although similar techniques could be used for tactical analysis. A simplified intelligence planning task was developed to demonstrate the concept of graphics modeling. In it, an intelligence performance model simulated,			

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

Block 20 (Continued)

the detection and classification performance of patrol units. A movement/terrain model guided user-postulated movements of units, and probability algorithms determined performance of units. The research showed that graphic simulations defined by the user can be developed to help conceive and evaluate potential combat activity.

The report is intended for military research and development personnel concerned with the application of interactive graphics to battlefield planning.

ACCESSION for	
NTIS	Office Section <input checked="" type="checkbox"/>
DDC	Section <input type="checkbox"/>
UNCLASSIFIED	<input type="checkbox"/>
SECRET	<input type="checkbox"/>
BY	
DISTRIBUTION/AVAILABILITY CODES	
Dist.	P. CML
A	

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

## FOREWORD

---

The Human Factors Technical Area is concerned with demands of the increasingly complex future battlefield for improved man-machine systems to acquire, transmit, process, disseminate, and utilize information. Research focuses on interface problems and interactions within command and control centers and is concerned with such areas as topographic products and procedures, tactical symbology, user-oriented systems, information management, staff operations and procedures, and sensor systems integration and utilization.

There is special research interest in the application of automated display systems to modeling and analysis of the dynamic battlefield. Promising modeling concepts, methods, and aids using effective graphic displays can enhance the division/corps command staff's ability to make timely and comprehensive interpretations and projections of battlefield information. The research described in this report provided methods for using interactive computer graphics to model combat events. It is part of an effort to cope with the modern battlefield's increased flow of tactical information by improving display formats and related analytic methods. Such research provides a necessary technological base for effective design of the user/systems interface.

Research in graphic analysis of tactical dynamics is conducted as an in-house effort augmented by contracts with organizations selected for their specialized capabilities and facilities. The efforts are responsive to general requirements of Army Project 2Q162722A765. They are related to special requirements of the U.S. Army Combined Arms Combat Development Activity, Ft. Leavenworth, KS, and the U.S. Army Intelligence Center and School, Ft. Huachuca, AZ. Special requirements are contained in Human Resource Needs 78-100 and 78-151.

**MODELING OF TACTICAL EVENTS BY INTERACTIVE GRAPHICS:  
APPROACH, INTERFACE DESIGN, AND SYSTEM DESIGN**

**TABLE OF CONTENTS**

<b>1.0 INTRODUCTION . . . . .</b>	<b>1</b>
1.1 Statement of the Problem . . . . .	1
1.2 Organization of the Report . . . . .	3
<b>2.0 METHODOLOGY FOR INTERACTIVE TACTICAL MODELING . . . . .</b>	<b>4</b>
2.1 Intelligence System Concepts . . . . .	4
2.2 Functional Flow of Selected Interactive Modeling Concept . .	15
2.3 Realism Aids and Support Models . . . . .	20
<b>3.0 CONCEPT IMPLEMENTATION . . . . .</b>	<b>35</b>
3.1 Man-Machine Interface Design . . . . .	35
3.2 Control Devices . . . . .	37
3.3 Intelligence Collection Planner Operation . . . . .	38
3.4 Intelligence Analyst Operation . . . . .	54
<b>4.0 EVALUATION AND RECOMMENDATIONS . . . . .</b>	<b>57</b>

## LIST OF FIGURES

1. Organization of Functional Flow for Collection Planning Application . . . . .	17
2. Functional Flow of Interactive Modeling Concept as Used by Intelligence Data Analyst . . . . .	21
3. Diagram of Major Design Features and Alternatives for Interactive Tactical Modeling . . . . .	23
4. Example of Local Interdistance Minima at $t_1$ , $t_2$ , $t_3$ , $t_4$ , and $t_5$ that are Input to the Detection Opportunity <sup>3</sup> Model . . . . .	33
5. Display Components . . . . .	35
6. Example of Terrain Defined by an Intelligence Collection Planner (ICP) . . . . .	42
7. Composition and Location of Opposing Forces at the Beginning of the Problem . . . . .	44
8. Positions of Intelligence Collection Foot Patrols 3:27 Hours into the Problem . . . . .	49
9. Positions of Enemy Forces 5:49 Hours into the Problem . . . . .	51
10. Positions of Opposing Forces 3:43 Hours into the Problem . . . . .	52
11. Alphanumeric Intelligence Messages and Symbols at 3:53 Hours into the Replay of the Problem . . . . .	55
12. Example of a Situation where the Existing Detection Opportunity Rule Produces a First Opportunity to Detect that is Unrealistically Late . . . . .	58

## LIST OF TABLES

1. Classification of System Concepts by Man-Computer Functional Allocation . . . . .	10
2. Evaluations of Combinations of Terrain Models, Movement Models, and Man-Machine Interface Approaches . . . . .	28

## 1.0 INTRODUCTION

### 1.1 STATEMENT OF THE PROBLEM

A central feature of planning doctrine in intelligence and operations is evaluation of alternative courses of action by wargaming. Traditionally, the command staff does battlefield planning and evaluation by "mental" wargaming and the use of hand-drawn map overlays to represent changing events. However, these manual procedures quickly become cumbersome, if not impossible, with the increasingly complex modern battlefield. In current warfare, the pace of action and numerous changing events combine to provide an enormous amount of information for the command staff to consider. Improved methods for representing and interpreting the flow of events on a battlefield are needed.

Combat simulation models provide a representation of warfare that could improve information utilization in planning and decision making tasks. As these combat models gain acceptance, the prime bottleneck will be coupling the user to the combat model — the user/model interface. Computer graphic systems are available to facilitate the coupling of combat models to the user. Easy graphics access to tactical models may allow the user to rapidly conceive, construct, examine and evaluate potential tactical activities. Such models exploit the use of user/computer interactive methods for aiding the command staff in tactical analysis and planning.

The project reported here developed a modeling methodology, implemented it in software on a computer graphics facility, exercised the software, and carried out a preliminary evaluation of the methodology. The focus of the research was on tactical planning although the same modeling procedures could be used in support of tactical analysis. The objective of the current methodology development and implementation was to demonstrate the graphics/modeling concept with a limited intelligence planning task.

The planning implementation uses the interactive graphics interface to define a tactical model of suspected enemy actions, define a collection plan for intelligence gathering units (patrols), and simulate their interaction. The resulting simulation provides an evaluation of the effectiveness of patrol plans, a limited section of the collection plan, and early and decisive detection of the particular enemy tactics represented by the tactical model. An iterative implementation of such a tool by the intelligence planner could allow the optimization of plans against one or more suspected enemy tactics.

The intelligence data analysis implementation consists of defining an enemy tactical model, simulating the interaction between the tactical model and the reported actions of the intelligence gathering units, and comparing the resulting simulated intelligence reports with those actually collected. Once again, an interactive implementation of this sequence would allow the intelligence analyst to determine the plausible tactical model of enemy actions that best explains the intelligence reports that were actually gathered. The resulting tactical model becomes a comprehensive interpretation of intelligence data.

## 1.2 ORGANIZATION OF THE REPORT

The remainder of this report is organized into Sections 2.0 through 4.0. Section 2.0 covers the evolution of the system concept of using interactive graphics to build tactical models using intelligence planning as an example. This starts in Section 2.1 with a functional breakdown of the intelligence process, identification of a number of system concepts as defined by man-machine functional allocation schemes, and an evaluation of these concepts with respect to predicted performance improvement and associated technological risk. Section 2.2 details a functional flow breakdown of the selected concept. Section 2.3 describes the algorithms and mathematical concepts used to implement the supporting models required by the interactive modeling concept.

Section 3.0 covers the operation of the concept as dictated by the specific software implementation developed in this project. The focus is on how the user takes advantage of the graphics functions to interactively develop tactical models. Finally, Section 4.0 presents a qualitative evaluation and some recommendations for future use and expansion of the concept.

---

Volume II, "Software Documentation and Algorithms (Appendices)", of this report is available from the U.S. Army Research Institute, 5001 Eisenhower Avenue, Alexandria 22333.

## 2.0 METHODOLOGY FOR INTERACTIVE TACTICAL MODELING

### 2.1 INTELLIGENCE SYSTEM CONCEPTS

Within the scope of the area being addressed, a limited number of intelligence functions must be performed to predict enemy actions and intent. A wide variety of associated functions will not be discussed since it was assumed that they are fixed within the system design assumptions and therefore they are not part of the problem. Functions to be discussed will not be assigned to either man or computer at this point, but will be defined in general terms. These functions were derived from three primary intelligence analysis considerations:

1. Tactical Hypotheses. What are the likely tactical plans of the enemy based on the experience of own forces commanders and their understanding of the tactical situation and the enemy's procedures?
2. Raw Intelligence Data. What are the actual raw reports and observations provided by the intelligence data gathering units?
3. Intelligence Gathering Performance. What reports are likely if a particular enemy action occurs, based on the performance capabilities and specific operations of own-forces intelligence gathering units?

Each of these considerations is crucial to intelligence analysis. If any of them is ignored, information is thrown away and prediction of enemy actions suffers. With regard to the first consideration, tactical hypotheses, a great amount of useful practical knowledge of likely enemy actions resides in the head of the field commander. At the present, such knowledge is poorly used by the computer aids provided for intelligence analysis. How to get such qualitative information into the computer in a form that can be incorporated into the computer aided intelligence analysis planning is a major problem that was addressed.

The second consideration, data presently available (or planned to be gathered by intelligence units and provided to the battlefield information computer system) is the foundation upon which other improvements must be built. It is in this area that most of the recent progress has been made, e.g., data coding schemes, management information/retrieval software, and raw data display features. However, additional improvement is possible and can be gained with the development of relatively simple algorithms operating within the framework of existing processor, storage, and display technology.

The third consideration, knowing what to expect from your intelligence gathering units, has also been severely neglected. What the observer units didn't report is at least as important as what they did report. Such data gaps only make sense when compared to expected enemy tactics and assumed intelligence gathering performance characteristics.

The following nine functions are considered necessary before the intelligence analysis/planning can be considered comprehensive. Each function can be carried out by either man or computer or some combination of both.

1. Intelligence Data Sorting and Subsetting

Each intelligence message consists of a number of characteristics (who, what, where, when, etc.). A fundamental function is the collection of groups of reports according to their similar or complementary characteristics. This function was previously handled manually but present system developments usually provide a computerized capability. This project assumed that such a capability is available in a general purpose management information/retrieval software package.

2. Intelligence Data Feature Selection and Trend Analysis

Just being able to group subsets of data will probably not provide the desired insight into enemy intentions. Additional operations and manipulation

(logical, mathematical, etc.) of the original intelligence data are necessary to investigate evidence of patterns of enemy activity. A crude example would be to call all the armor sightings within a given time slice a "cluster" and then solve for the centroid of the cluster. Time based movement of such a centroid might give an indication of a shift in the strength of the enemy's forces.

A human is capable of performing some of these operations relatively well and envisioning the result mentally. He does have the advantages of (a) real time reprogramming of his mental "software" and (b) a variety of sometimes-sophisticated, non-linear weightings for individual data points. However, the machine provides accuracy, repeatability, objectivity, and tirelessness.

### 3. Intelligence Data Display

As stated previously, it is assumed that a graphics display capability will be available. The display functions provided are then dependent upon the display software characteristics. This function refers to the ability to display the results of functions 1 and 2.

This function could be provided by a human by mentally visioning replays of data maps. However, since it has been assumed that the data is in the computer and the computer drives a graphics display terminal, it will also be assumed that this function is allocated to the computer.

### 4. Enemy Tactical Model Storage

A tactical model defines the pertinent movement and actions of an enemy force's units. Such a model becomes a testable hypothesis that can be used as an intelligence analysis/planning aid.

A knowledgeable field commander can be expected to generate alternative enemy tactical models. To expect a computer to sift through the nearly infinite set of feasible tactics and select a few meaningful

alternatives is not within the state-of-the-art of field-mobile systems and their associated software. However, a tactical model conceived by a human could be put into computer storage for later use. Thus, this function refers to what happens to the tactical model information once it has been conceived by a human. If this function is to be provided by a computer, it must include not only the storage capability but also the man-machine interface which allows the translation from what is in the field commander's mind to a computer coded form. If this function is allocated to the human, he must conceive, remember, and be able to replay in his mind each of the alternative hypothetical enemy tactics.

#### 5. Tactical Model Feature Selection and Trend Analysis

The hypothesized tactics and the raw data are amenable to mathematical analysis. Applying information reduction and simplification procedures to both the raw data and a hypothesized tactical model should facilitate quick and easy comparison between data and model.

If this function is allocated to the human, he must envision the results of such operations on the hypothesized tactical model in the same manner as he would for the raw intelligence data. This most likely will take the form of "visual" transformations of envisioned dynamic tactical maps being replayed in the mind.

#### 6. Simulation of Own-Forces Intelligence Activities

The simulation would use knowledge of the capabilities and specific activities of own-forces intelligence gathering units, and would allow investigation of what intelligence data should have been observed if any given enemy tactic were actually being implemented.

If this function is allocated to the human, he must "normalize" the raw data for the terrain covered by mobile intelligence gathering units. This might be carried out by imagining a dynamically changing portion of

the local terrain being searched and the corresponding data that should or should not have been gathered as a result of what the enemy was doing.

#### 7. Display of Own-Forces Intelligence Activities

If a record of the activities of the intelligence gathering units has been put into the computer and stored, it then can be replayed via the display unit. This function not only assumes the display function but also the man-machine interface necessary to enter such information into computer storage.

#### 8. Enemy Tactical Model Display

If the information for a hypothesized enemy tactical model has been entered into the computer, it can then be replayed by the display for visual inspection and analysis.

#### 9. Enemy Tactical Model Testing

Comparison of the raw or analyzed intelligence data with what would be expected from a combination of intelligence gathering activities and enemy tactics is the final function. It is here that the bottom line decision is made concerning which of the alternative hypothesized enemy tactical models (or some combination thereof) is presently being or soon will be implemented by the enemy.

If the function is allocated to the human, he must use what partial display functions are available and envision the rest. He must make a statistical decision concerning the likelihood that the present intelligence data base would have resulted given the enemy was carrying out a given tactic. Allocation of this function to the computer poses questions of mathematical sophistication and data dimensionality of staggering proportions. Overcoming such obstacles is a topic that must be addressed by a major effort program. The modest effort carried out by this project produced a small but significant first step toward such a major program.

Within the assumptions and constraints previously described, an intelligence data analysis/planning system concept can be defined by allocating each of the nine functions to the man, or the computer. Not all possible combinations are meaningful or balanced.

Table 1 shows six different system concepts selected because they were felt to be meaningful increments in a gradually increasing level of automation (A through F). The table consists of functions (rows) by system concepts (columns), with the cells indicating whether the particular concept had a given function allocated to a human decision maker (H) or the computer (C). The bottom row of the table shows the number of functions allocated to the computer. This is a crude measure of level of automation embodied in the system concept.

Several features about the table should be identified before individual system concepts are discussed. As previously mentioned, some functions are always assumed to be allocated to the computer as far as the scope covered in this work is concerned. These are: "1. Intelligence Data Sorting and Subsetting," and "3. Intelligence Data Display." Also, no display function can be allocated to the computer unless those functions necessary for the information to reside in the computer have been also allocated to the computer.

System Concept A, with only functions 1 and 3 allocated to the computer, is very similar to many existing systems (Navy NTDS, etc.) where selective display of subsets of alphanumeric data is all that is provided. A great potential increase in cost-effectiveness can result from research in data coding, data file structures, associative memories, color-coded display format, data file trace routines, and many other areas.

System Concept B adds the intelligence data analysis (2) feature to those automated in System Concept A. This system concept is similar to that envisioned in "Computer-Based Displays As Aids in the Production of Army

**Table 1. Classification of System Concepts by  
Man-Computer Functional Allocation**

<p>H - Function Carried Out by Human C - Function Carried Out by Computer</p>			<b>Intelligence Analysis/Planning System Concepts</b>					
			A	B	C	D	E	F
Intelligence Data Analysis Functions	1	Intelligence Data Sorting and Subsetting	C	C	C	C	C	C
	2	Intelligence Data Feature Selection and Trend Analysis	H	C	H	C	C	C
	3	Intelligence Data Display	C	C	C	C	C	C
	4	Enemy Tactical Model Storage	H	H	C	C	C	C
	5	Tactical Model Feature Selection and Trend Analysis	H	H	H	H	C	C
	6	Simulation of Own Forces Intelligence Performance	H	H	H	H	H	C
	7	Display of Own Forces Intelligence Activities	H	H	C	C	C	C
	8	Enemy Tactical Model Display	H	H	C	C	C	C
	9	Enemy Tactical Model Testing	H	H	H	H	H	C
		Number of Functions Automated	2	3	5	6	7	9

Tactical Intelligence."\* System Concept C concentrates on being able to input all three information types (raw intelligence data, hypothesized enemy tactical models, and intelligence gathering unit activities) into the computer and recall them for various types of dynamic display formats. This uses the computer as an information entry, storage, and replay device with the human performing mental processing and looking for data patterns indicative of enemy tactics.

System Concept D automates the feature of intelligence data analysis. It is important to distinguish between intelligence data analysis and analysis which considers two or all three (intelligence data, tactical hypotheses, intelligence activities) types of information. The latter does not occur until System Concept F. System Concept E automates the same functions as System Concept D in addition to intelligence data analysis procedures on the hypothetical enemy tactics (function 5). This allows the same operations to be carried out on the intelligence data and hypothesized tactics and the respective results to be replayed on the display simultaneously. This would greatly improve the likelihood of the human perceiving similarities between data and tactical models. System Concept E would also allow simultaneous replaying of intelligence unit activities for visual inspection but would not permit any mathematical operations to be done on this information.

System Concept F adds two automated features to those included in System Concept E. The first is to allow mathematical operations on intelligence unit activities and information by means of a computer-based simulation of predicted performance. The second feature (function 9) is the most powerful, complicated, and futuristic yet mentioned. It provides the ability to tie all three types of information together in a single mathematical operation leading to a statistical conclusion about the likelihood that the existing intelligence data resulted from observing any given hypothetical

---

\* U.S. Army Research Institute for the Behavioral and Social Sciences, Technical Paper 258. Bowen, Russel J. et al. February 1975.

enemy tactical model. There are a variety of mathematical procedures which hold promise for implementing such a concept, but a pattern recognition/cluster analysis approach would seem most compatible with the data.

System Concept A (Table 1) presently exists with refinements already being researched. System Concept B has also had some preliminary research carried out on its unique feature, computer aided intelligence data trend analysis. System Concepts E and F are quite advanced, and preliminary investigations are necessary before they can be considered. Fortunately, the preliminary considerations for Concepts E and F and the unique features of Concept C are the same. Consequently, System Concept C was selected to provide the framework within which the research project operated. The following unique features were investigated with varying degrees of emphasis:

- Man-Machine Interface (MMI) Design for Entry and Replay of Hypothetical Enemy Tactical Models
- Decision Aids to Evaluate the Realism, Consistency, and Viability of Hypothesized Enemy Tactical Models
- Use of Simulations of Intelligence Unit Activities for Intelligence Data Interpretation
- Interactive Graphics for Intelligence Data Interpretation

Each of these four features will be discussed in expanded detail (with examples) in the remainder of the report. The main emphasis of the research project was on the MMI for entry and replay of enemy tactical models. Simulations of intelligence units and decision aids for evaluating the realism of hypothesized enemy tactics were explored at the minimum level required to facilitate the primary thrust. Interactive graphics was used at each step and as such is not a separable feature.

The essence of man-computer symbiosis is to take maximum advantage of the experienced field commander's insight, judgment, and qualitative information while allowing the computer to unburden him from tedious, time consuming tasks. This not only provides the man-machine team with the

greatest amount of information, but maximizes the acceptability of the hardware system to the user because it converses in his terms and allows him to use his rationale. The research topic of inserting hypothetical enemy tactical models into the computer for later analysis and display replay is aimed at achieving this essence.

In any tactical situation a variety of tactics are possible and often even reasonable. However, most field commanders can reliably reduce the thousands\* of feasible tactics down to five\* or fewer that are most likely. This is a talent that no computer system presently possesses. How does the system designer take advantage of this talent? It is necessary to provide the essential details of any suspected tactical model to the computer. It is likely that an interactive graphics display is the most fluent translator for this man-computer communication.

The procedure for entering a hypothetical enemy tactical model consists of four basic steps:

1. Identify characteristics (type and size) of enemy units.
2. Determine a time-based movement path for each enemy unit.
3. Determine a time-based activity schedule for each enemy unit.
4. Repeat steps 1 through 3 to define a coordinated action.

The capabilities to carry out these four steps via an interactive graphics interface were incorporated in the software implementation of the interactive tactical modeling concept.

Once the elements of a coordinated hypothetical enemy tactic have been entered into the computer, the information can be used in a variety of ways. One is simply to replay on the screen the complete coordinated tactical scenario. The tactical model can be used in a simulation to produce examples of intelligence data that are likely to be gathered. Regardless of the

---

\*Obviously very rough numbers which can vary drastically from case to case.

ultimate use, the ability of a field commander to input his hunches about the enemy's actions into a computer and be able to exercise those assumptions in a variety of ways is a promising concept deserving of extensive exploration.

While the ability to input hypothesized enemy tactics into the computer is a powerful concept, its impact on intelligence data interpretation is only as good as the tactical model developed by the field commander. Even though the experienced commander may well be the best source of suspected enemy intentions and propensities, he can use considerable help in ensuring that the details of his hypothesized tactical model are realistic and consistent. The primary elements of hypothesized tactical models are units (type and size), movement of the units, and actions of the units (e.g., a given artillery unit shelling a certain area of the terrain). The real world places constraints on each element individually and as a function of their plausible coordination with each other. Terrain, weather, road type, vehicle type, etc., place limitations on reasonable speed-made-good for travel from any point to any other point. Tactical doctrine places reasonable limitations on how different types of units coordinate with each other (e.g., armor usually precedes infantry). Knowledge of equipment places constraints of events or actions of enemy units, e.g., the allowable terrain reachable for shelling by a given artillery unit from a given position. Consequently, movement and firing realism aids were incorporated into the software implementation of the interactive tactical modeling concept.

One of the crucial errors that can be made in interpreting intelligence data is assuming that it was collected by a process that uniformly covered the terrain of interest. This is never the case and often a group of independently operating intelligence units will put a very unique set of biases into the intelligence data collection effort. Even a crude simulation of the activities of the intelligence units, once put into the computer, can greatly improve interpretation of data. The basic performance models of

observation (detection, classification, etc.) can be developed before taking the field. The actual activities of foot patrols, drones, etc., can be put into the computer in much the same manner as for tactical models. Once in the computer, the simulation can be used to derive a graphics display in replay mode or to produce statistical data in simulation mode. A performance model of the detection and classification capabilities of the simulated intelligence units was developed and implemented into software. The model was constructed such that terrain types along the line-of-sight between an intelligence unit and potential target had an appropriate effect on probabilities of detection and correct classification.

Finally, a compressed time replay on a graphics display capability is useful from a variety of standpoints. The ultimate use, of course, is to replay a given tactical model against a proposed collection plan or real-time intelligence data. This allows visual inspection for meaningful correlative patterns between tactical model features and actual reported intelligence messages. Such a replay capability is also extremely useful in the tactical model development stage. To be able to review a partially completed tactical model in order to insure that the next input results in a coordinated tactic is crucial to the success of interactive modeling. The interactive compressed time replay of complete and partial tactical models was included in the software implementation of the concept.

## 2.2 FUNCTIONAL FLOW OF SELECTED INTERACTIVE MODELING CONCEPT

The decision system within which the user/model graphics interface must be contained consists of three primary elements: man, hardware (computer and display), and software. The goal of the interface design task was to use the respective strengths of each element in a manner that compensated for weaknesses in other elements. The man and hardware elements came with fixed capabilities (sometimes fixed in a statistical sense) while the software had the greatest flexibility. The crux of the problem then became how to use the software's flexibility in a manner that took advantage of the man's insight into the real tactical and the computer's speed, accuracy, and reliability.

The approach focused on taking advantage of man's inherent talent as a "picture-thinker" and his ability to envision tactical alternatives. Remaining in the picture medium, the interface was designed to allow the user to communicate tactical models in terms of pictures via an interactive graphics terminal. This eliminated the need for any extensive knowledge of either computers or math models on the part of the user. The military user can continue to think and communicate tactical considerations in the medium in which he is trained: maps and pictures.

The software was implemented on a four-color, refresh, vector graphics terminal. A number of unique algorithms were developed to enable graphical input of tactical models; these will be introduced in Section 2.3. Much thought was placed during organization of the software so that it would have greatest flexibility of use. Consequently, the resulting software can be also used for tactical planning research as well as intelligence aids.

The following functional flow analysis was used to identify the functions, support models, and interactive capabilities required for implementation of the concept. It was also used to guide the hardware/software functional allocation incorporated into the implementation design effort. As it will be shown in Section 3.3.1, the flow of information developed in this analysis had a substantial impact on design of the MMI and support models used in the concept implementation.

The functional flow analysis was carried out for both of the two primary users: collection planner and intelligence analyst. Eventually, the software implementation resulted in one software package with the two users merely using different operational modes. All the major functions and support models were implemented with both modes.

The organization of the major functions is shown in Figures 1 and 2. Figure 1 shows the flow when the concept is being used in a Collection Planning mode, and Figure 2 shows the flow when the concept is used by the

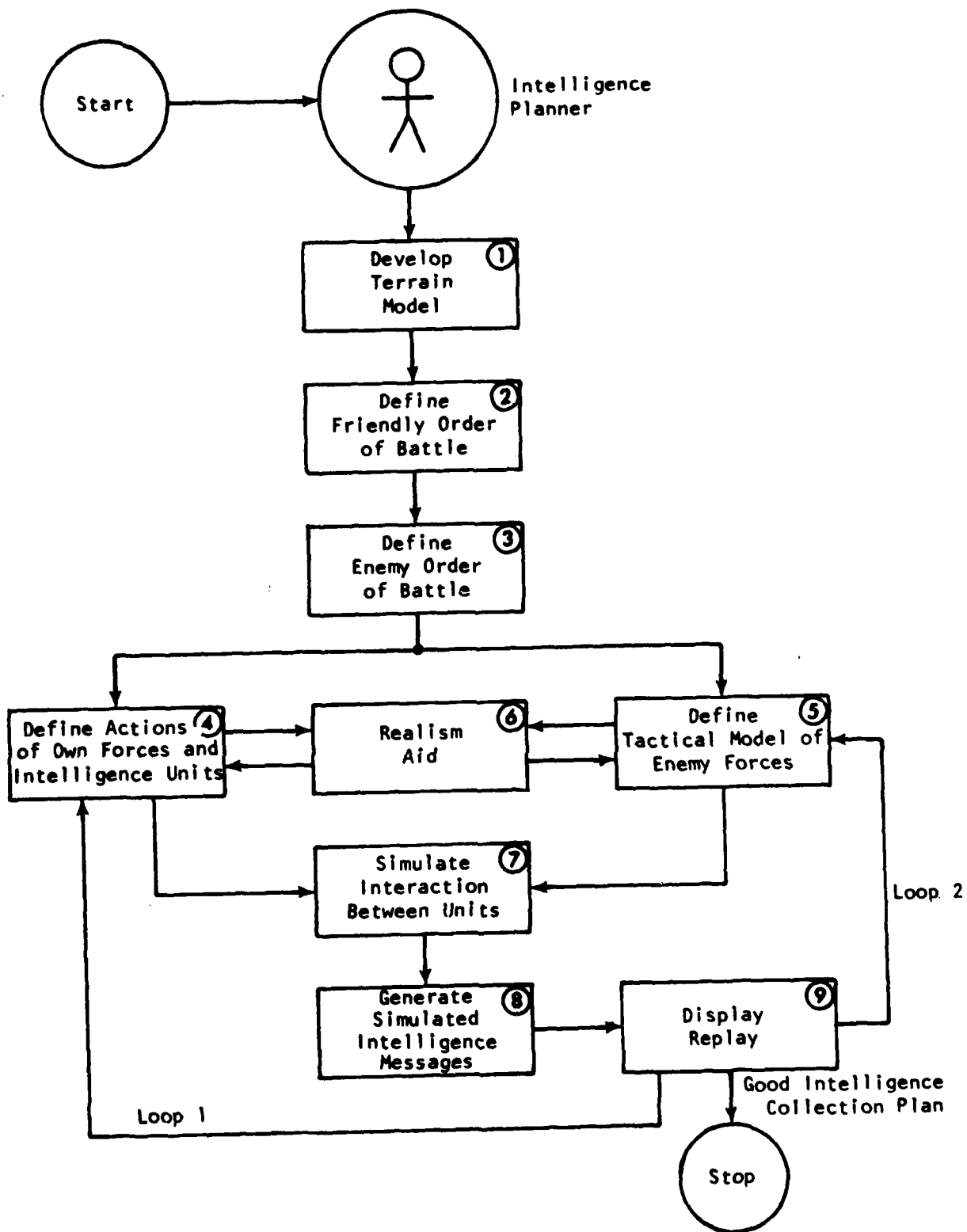


Figure 1. Organization of Functional Flow for Collection Planning Application.

Intelligence Analyst for intelligence interpretation. Each box in the figures represents a major function, and boxes numbered the same in both figures represent the same identical function.

As Figure 1 depicts, the collection planning application is initiated by the scenario development which consists of functions 1 through 3. Function 1, terrain model development, uses the interactive graphics interface to allow the user to input a terrain model via the display. Such terrain features as cities, mountains, lakes, forests, roads, and rivers are defined in the computer by the user actions on the display. Functions 2 and 3 are identical but create different data bases; they consist of the definition of the respective order of battle of the two opposing forces. This includes defining the initial positions of each unit (e.g., company) on the just-created terrain map geographic display.

Functions 4, 5, and 6 are used to create the dynamic movement, action, and interaction of the two opposing forces. Again, functions 4 and 5 are identical except for different data bases. They consist of using the graphics interface to define the movement and actions of all the units of the opposing forces. In the case of the collection planning application, it is assumed that the planner knows friendly forces' actions, must predict enemy actions, and is interested in exploring the relative effectiveness of various actions of the intelligence gathering units. A major function in this phase is the realism aid (box 6) which provides feedback to the user on the validity of movement rates over various terrain features and other actions such as direct and indirect fire orders.

Once the tactical models of the two opposing forces are complete, these data bases are provided to function 7 which simulates their interaction on a time-phased basis. The simulation focuses on the generation of intelligence messages instead of combat results such as attrition rates. Thus function 7 combines a movement model with the terrain model developed in function 1. Function 7 also includes a line-of-sight model which keeps

track of the distances and intervening terrain features between each friendly intelligence unit and each enemy unit. As the scenario is played through, opportunities for intelligence sightings are recorded. Times and positions of the opposing units are sent to function 8 when these opportunities occur.

Function 8 consists of the detection and classification performance models of the various intelligence gathering units. Each opportunity recorded in function 7 is simulated. If a detection occurs, the classification model is exercised and a simulated intelligence message results. The complete set of intelligence messages then forms a time-correlated data base available for animated viewing.

Function 9 consists of a compressed-time replay of the intelligence message in graphical and alphanumeric form. The collection planner will use this replay capability to investigate the predicted effectiveness of his proposed collection plan (as defined by the previously-input movement of intelligence units) against the suspected enemy tactic (as defined by the previously-input movement and actions of enemy units).

At this point the collection planner's use of the software forms two nested iterative loops. Loop 1 uses the feedback of the displayed information to refine the collection plan and resulting performance against a given enemy tactic. Loop 1 is nested within the outer loop (2). Loop 2 consists of investigating collection planning effectiveness against the various tactics that the enemy might reasonably try to implement. In this manner, a collective plan which is optimally effective against the most likely enemy tactic but robust enough to do a good job against a variety of likely enemy tactics should result.

The concept as implemented via software for the collection planner can also be used by an intelligence data analyst in the field, training, or research. If used for training or research, functions 1 through 9 can be used to develop a "true" enemy tactical model which the intelligence

analyst attempts to decipher. This "true" model is used to generate the simulated intelligence messages upon which the analyst uses the concept (functions 5 through 9) to develop his estimate of the enemy tactical model. In the field, of course, the intelligence messages come from the actual enemy actions, but the use of the concept from that point on is identical.

Figure 2 shows the functional flow of the concept as used by the intelligence data analyst. The intelligence analyst uses the system to view the data he normally would have access to, namely, movement and actions of own forces and intelligence units and reported intelligence messages, to attempt to decipher the correct enemy tactical model. The intelligence analyst uses functions 5 through 9 to develop his predictions of the tactics the enemy is carrying out. He then compares the displayed interactions of the intelligence units and enemy units and evaluates as to whether they could explain the intelligence messages that were "received." He iteratively modifies his tactical model of the enemy until it appears to explain not only the messages that were received but also those that were not. In this fashion the intelligence analyst uses the interactive graphics procedure to define an enemy tactical model that forms an interpretation of a set of intelligence data.

## 2.3 REALISM AIDS AND SUPPORT MODELS\*

### 2.3.1 Evaluation of Competing Support Models

The concept that was developed and implemented was a modeling procedure, not a model per se. However, in order to implement the interactive modeling procedure in a valid, realistic manner, a number of models and realism aids were required. These models are referred to as support models because they support the modeling procedure. The evaluation of various

---

\*This section, and especially Section 2.3.1, is aimed at persons with knowledge of man-machine interface design and mathematical modeling. Persons whose main interest lies in understanding how the concepts were implemented may wish to proceed to Section 3.

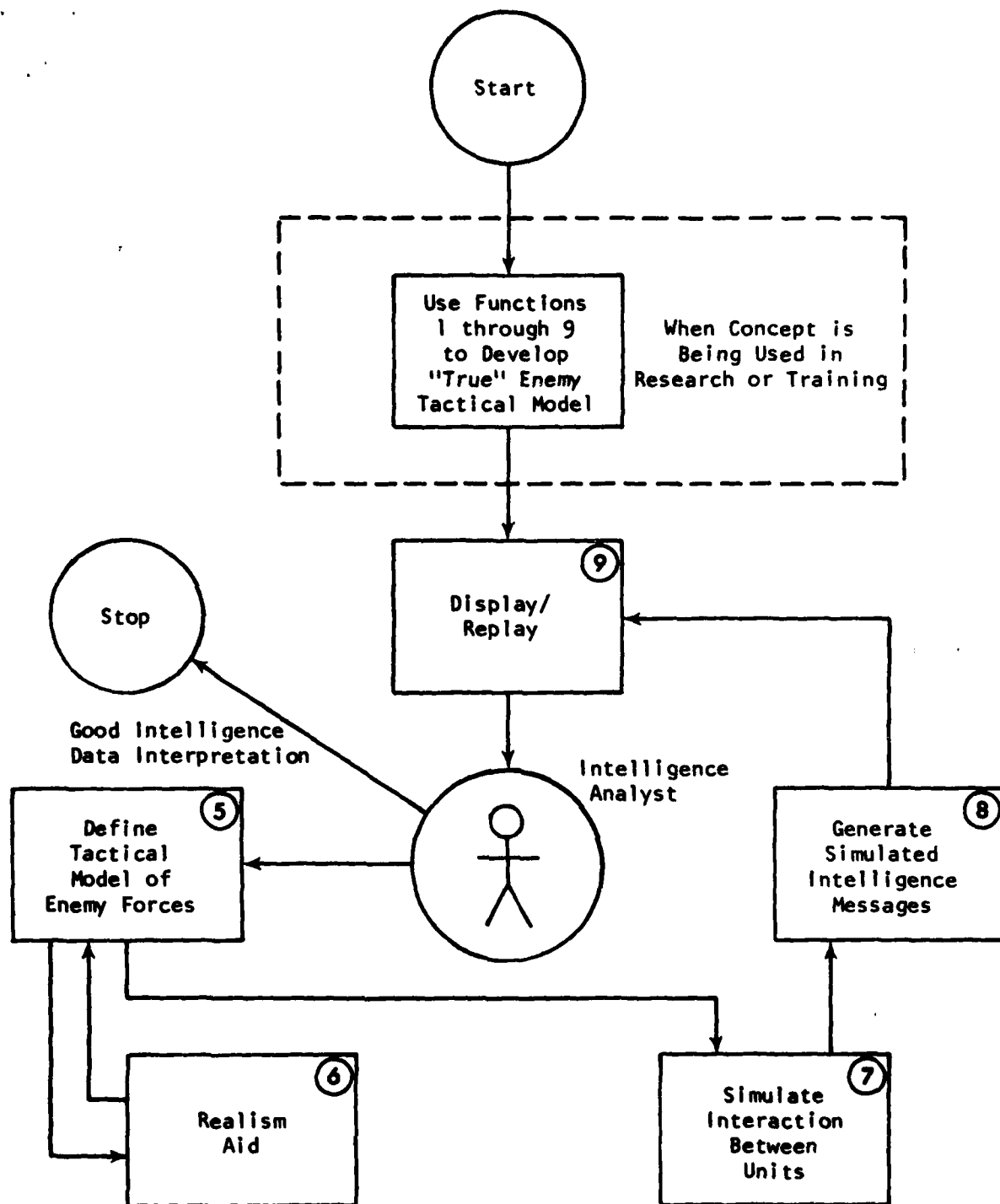


Figure 2. Functional Flow of Interactive Modeling Concept as Used by Intelligence Data Analyst.

alternative algorithms for implementing a given support model was confounded by critical design features of the man-machine interface. Thus, the selection of support model algorithms and the design of the interactive modeling MMI was approached as an integrated design problem.

Figure 3 depicts the five major design decisions and their respective alternatives. The combinations of the five design decisions--terrain model, movement model, movement dimension, control feedback, and replay feedback--yielded 72 alternative approaches.

The terrain and movement models are two of the most crucial in their impact on the viability of the concept. The terrain model is used for a variety of purposes, the most important of which is to define allowable movement rates from any given point to any other given point. Other purposes include determining line of sight versus occlusion for intelligence units and defining allowable regions for direct and indirect fire of weapons. Two main approaches were identified for defining a computer-based terrain model: discrete versus continuous. There are two subcategories within the discrete: grid oriented and contour oriented. A discrete terrain model is one in which features of the terrain, such as altitude or allowable movement rate, are discretized according to a geographically based grid. Thus the terrain model itself consists of a data file, each number representing the feature value for a given geographical grid cell, e.g., hexagonal shape, 300 meters across. The continuous terrain model, on the other hand, can identify a feature value for any possible point within the geographical area of interest. Thus the continuous terrain model is stored in the form of coefficients which define the appropriate functional equations.

Within the discrete variety there are two approaches to storing the same information within the computer. The grid oriented approach is the most straightforward and simply divides the geographical area into an array of small areas and stores a feature value for each grid cell. The contour approach uses graphically drawn contours to identify domains of equal



feature values. For example, a city would have a contour drawn around it, indicating that all of the grid cells within the contour contain the movement feature value for city traffic. Using this form of terrain model, the computer must interrogate the contours each time it needs to identify the feature value of a particular grid cell.

Each of the terrain model approaches has its relative strengths and weaknesses. The three main dimensions with which the competing terrain model approaches were evaluated are: core size required, speed of access, and impact on validity of use. The continuous approach is best with respect to core size in that only a few (e.g., less than 100) parameters have to be retained in order to provide the model. The continuous approach is likely to be faster than the discrete contour approach, but slower than the discrete grid approach. A lot depends on the functional forms used; polynomials would be faster than exponentials. While continuity certainly impacts validity in a positive way, the need to deal with a limited set of functional forms may degrade terrain model validity. Thus, the continuous approach appears to be less flexible than the discrete approaches. Finally, the continuous approach may simplify the logic involved in its use.

The grid-oriented approach is the most intensive in core requirements, e.g., multiple thousands of numbers. It is the quickest in speed of access and can be the most flexible and therefore most valid. The ability of the grid oriented approach to be valid depends on the level of resolution, which is inversely related to core requirements. An additional consideration which impacts the cost effectiveness of the grid-oriented approach is the labor consuming task of entering the grid data array into the computer.

The contour-oriented approach is medium in core requirements, e.g., hundreds of numbers. It is slower than the grid in access, but may be faster than the continuous. It is less valid than the grid-oriented approach in that it is less flexible and allows fewer alternative values for any feature dimension. However, it is most convenient to build a

contour-based terrain model by interactive graphics and therefore encourages the easiest changes to the terrain model when required.

The movement model determines how rapidly a given vehicle can traverse a given path through the terrain model. Thus, the movement model must access terrain movement allowance information from the terrain model and then determine the position of a vehicle along a path as a function of time. There are two types of movement models corresponding directly to the two types of terrain models: discrete versus continuous. There are three identifiable alternative dimensions of the movement information stored in the terrain model: velocity, time, and movement allowance.

Time refers to the time required to allow a given vehicle to pass from one point to another of a given terrain type. For a given path and a given vehicle the time dimension would consist of the earliest or nominal arrival time at the end of the path. Velocity can refer to either the nominal or maximum allowable velocity for a given vehicle while operating within a given terrain type. Movement allowance is an idea borrowed from the war games that are commercially available. In these, time is chopped up into plays and phases of the game; each vehicle has a movement allowance that it can expend within a given play. Each terrain then has a movement cost, and that number of movement allowance points is used up by transiting through a cell of that terrain type. This approach could be implemented on the computer as well. The compatibility of each of the three variables (time, velocity, and movement allowance) differs with respect to the two terrain models (discrete and continuous).

Thus, a movement model uses the movement information incorporated in the terrain model, matches it with a proposed geographical path, and calculates the time at which a vehicle going over the proposed path will arrive at each point on the path. The result is a distance progressed along the proposed path as a function of time information. If the terrain model is discrete, this information is also discrete; it consists of the beginning and ending times that the said vehicle spends within each grid resolution

cell intersected by the proposed path. If the terrain model is continuous, the result is a continuous curve defining distance progressed along the path by the vehicle as a function of time. In the discrete case, the calculations involve simple arithmetic, while in the continuous case, the calculations involve integral calculus or numerical approximations thereof. The selection of a movement model cannot be made independently of the selection of either (a) the terrain model or (b) the man-machine interface by which the operator uses the movement model to define and evaluate proposed movements of enemy units.

The man-machine interface design for accessing the movement model to determine a unit path profile can be described by two dimensions: type of control feedback and type of replay feedback. Control feedback refers to the feedback display on the controlled variable, e.g., velocity as a function of distance along path. There are two types of control feedback: direct and indirect. This is true whether the control variable is time, velocity, or movement allowance (distance). *Direct control feedback* consists of the presentation to the operator of a geographical map and the selected path of unit movement. The operator manipulates a control device to vary the controlled variable, and the resulting feedback consists of displaying unit(s) moving along the path in the geographical presentation. *Indirect control feedback* consists of a functional plot of a controlled variable versus distance along path in a classical two-dimensional, x-y plot fashion. The direct control feedback approach provides geographical information, but such things as rates along a path may be difficult to perceive or remember. The indirect provides such information as velocity profiles directly for the observation of the user but requires that the operator go back later to evaluate the impact on the actual geographical movement.

Replay feedback consists of presentation of the latest movement information input by the observer being replayed with all the other movement information available. This is necessary in order for the observer to insure that he has created a coordinated tactic. There are two fundamental

approaches for replay feedback: simultaneous and iterative. Simultaneous replay feedback refers to the case when the state of the complete tactic involving all units is advanced at the same time as the unit presently being controlled. This allows the operator to get the most instantaneous feedback on coordination of various units while he is determining the movement of a given unit. Iterative feedback refers to a sequencing of control feedback versus replay feedback. In this approach, the user would see only the motion of the unit being controlled during control feedback and then would go to a replay mode where he would then see that motion interacting with the motions of other units. Simultaneous replay feedback provides the most information; it may demand too much processing for a given level of tactical complexity. The processing demands may lead to unacceptable blinking on the display or may not be feasible at all for certain types of terrain and movement models. The iterative replay feedback is processing-economic but more demanding on the operator's memory.

Table 2 illustrates the combinations of terrain models, movement models, and man-machine interface approaches which result from considering the previously identified alternatives. Within the table, a given cell represents a system concept resulting from the combination of a single terrain model, a single movement model, and a single man-machine interface approach. Each cell of the table is coded with respect to a coarse preliminary evaluation, namely, low, medium, and high. The information in the table should not be scrutinized in detail but rather investigated for meaningful patterns. For example, the row representing the combination of indirect control feedback with simultaneous replay feedback is rated low because of the basic incompatibility: it doesn't make sense to bother with simultaneous replay feedback when there is only indirect control feedback. Also, the columns associated with applying a discrete movement model to a continuous terrain model are all evaluated low because it doesn't make sense to throw away the high resolution information in the continuous terrain model by applying the more crude discrete movement model to it. Any row in which movement allowance is the control variable for a continuous movement model is rated low. Again, the movement allowance concept was developed

Table 2. Evaluations of Combinations of Terrain Models, Movement Models, and Man-Machine Interface Approaches.

TERRAIN MODEL		DISCRETE GRID						DISCRETE CONTOUR						CONTINUOUS					
MOVEMENT MODEL		DISCRETE			CONTINUOUS			DISCRETE			CONTINUOUS			DISCRETE			CONTINUOUS		
CONTROLLED VARIABLE		TIME		VELOCITY		MOVEMENT		TIME		VELOCITY		MOVEMENT		TIME		VELOCITY		MOVEMENT	
		REPLAY	FEEDBACK	INTER-ACTIVE	SIMUL-TANEOUS	ITERATIVE	SIMUL-TANEOUS	ITERATIVE	SIMUL-TANEOUS	ITERATIVE	SIMUL-TANEOUS	ITERATIVE	SIMUL-TANEOUS	ITERATIVE	SIMUL-TANEOUS	ITERATIVE	SIMUL-TANEOUS	ITERATIVE	SIMUL-TANEOUS
MAN-MACHINE INTERFACE		INDIRECT		DIRECT		INDIRECT		DIRECT		INDIRECT		DIRECT		INDIRECT		DIRECT		INDIRECT	
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
		19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36
		37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54
		55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72

KEY:

Low

Medium

High

for discrete movement models and is a mismatch with continuous movement models. The evaluations of the other cells were arrived at by brief consideration of the compatibilities between the respective terrain model, movement model, and man-machine interface approach. As a result of this brief investigation of system concepts, roughly two-thirds were evaluated low, one-sixth medium, and one-sixth high.

Since the desired features of the man-machine interface design were so dependent upon the selection of the terrain and movement models, these decisions were made first. The selection of the terrain model was made first. The continuous terrain model was eliminated because of low validity, mathematical complexity of use, and high processing costs of use. The discrete, grid-oriented terrain model was eliminated because of high core requirements, difficulty in varying the terrain for test purposes, and incompatibility with the desire to implement a geographic zoom feature in future work. Consequently, the discrete, contour-oriented approach was selected because of its maximum flexibility, moderate core requirements, moderate processing requirements, and moderate validity.

With the selection of the terrain model, the relative desirabilities of the movement models became more clear. The contour-oriented terrain model allows the more desired continuous movement model, so it was selected. Thus, the combination of the contour terrain model and continuous movement model provides the greatest flexibility and allows a complete geographical zoom to be added quite easily at a later date. The availability of continuous motion of the military units had a direct impact on the availability and desirability of control variable and replay feedback options. The continuous motion provides the degree of resolution needed to allow direct control of the motion variable. However, continuous motion does require considerable processing and diminishes the tractability of simultaneous replay feedback. Consequently, direct control of the motion variable and iterative replay feedback were selected.

The remaining major selection to be made in the design of the man-machine interface was the selection of the controlled-motion variable. The movement allowance variable was eliminated by the selection of the discrete contour terrain model. The choice between the remaining two variables of time of arrival and velocity was difficult. It was crucial to keep the purpose of motion control in mind, namely, to develop a hypothetical tactical model consisting of the coordinated motions and actions of a number of enemy units. The key word is coordinated. In the initial system, the intelligence planner or analyst can define the motion of only one unit at a time. But he must do so in a manner that each unit's motion is matched to the master plan and each other unit's motion. He must create meaningful patterns of unit positions at crucial points in time, and he must avoid unrealistic patterns for all points in time.

If the velocity dimension were chosen as the motion variable, the analyst would be faced with attempting to define a time varying velocity profile for each unit so that it arrived at the right place at the right time. This would be relatively difficult, particularly when one considers the impact of unit/vehicle type and terrain features on allowable velocities. It was much simpler and effective to use time of arrival of each unit at each pertinent geographic position directly. Such an approach is only usable with a realism aid controlling minimum arrival times as determined by unit type/terrain type matchups. However, the contour-oriented terrain model is well adapted to such movement realism aids.

In conclusion, a system concept was devised consisting of a terrain model, movement model, and man-machine interface design. A contour-oriented, discrete terrain model was selected because of its maximum flexibility and moderate core requirements, processing demands, and model validity. A continuous movement model was selected for validity and geographic zoom enhancement. The man-machine interface was selected to consist of direct control and iterative feedback. Direct control was selected because of its compatibility with continuous motion, while iterative feedback was necessitated by the processing demands of continuous motion. Finally, unit motion

was determined to be best controlled by time of arrival as opposed to movement allowance or velocity.

### 2.3.2 Terrain Model

The contour-based discrete terrain model consists of piecewise linear contours and associated terrain type labels. The computer data storage of this terrain model consists of the beginning and ending points (in x,y coordinates) of each linear contour segment. Thus, the geographic area of interest has a variety of closed and open contours defined upon it. The closed contours identify the terrain type as existing everywhere within the contours. Cities, forests, hills, and lakes are examples of terrain types defined by closed contours. The open contour terrain types are defined to exist only on the straight-line segments themselves and not in an area. Rivers and roads are examples of open contour terrain types.

Thus, the system user defines various terrain types by "drawing" their respective contours (opened and closed) on a graphics display terminal. This data is entered in the computer in straight-line endpoint format and resides there for future accessing. A terrain type priority table is used to resolve conflicts that arise when any given point in the geographic area of interest is overlapped by two or more terrain types. In this fashion, any given point in the geographic area can be identified with a unique terrain type.

Each terrain type has a number of feature values associated with it. The two crucial features are mobility and visibility. Each terrain type has values for the nominal and maximum velocity that each unit type can attain in transiting through that terrain type. These values are stored in the mobility table. The impact of a given terrain type intervening between an intelligence unit and potential target is reflected by values stored in the visibility table.

### 2.3.3 Movement Model

The movement model combines the user-defined paths of motion for a given unit with the terrain model to generate earliest and nominal times of arrival at points along the path. The length of the segments of the path spent in each terrain type is measured and a constant velocity is assumed throughout a terrain type. Any velocity less than the maximum can be selected by the user. Once a user has selected acceptable unit velocities for each path segment, the progress of the unit at any point along the path can be determined as a function of time.

In this fashion the terrain model and associated mobility table provide the function of a realism aid. The user is prevented from building tactical models consisting of unrealistic movement rates of certain units over certain terrain types. This also provides the user feedback on what are likely tactics for the enemy to try to implement within the local terrain.

### 2.3.4 Performance Models for Intelligence Units

A crucial support model is the one which predicts the performance of the intelligence units. This model must consider the relative time-varying distance between each intelligence unit and potential target and the intervening terrain types.

The intelligence performance models are exercised in four phases. Phase I consists of identifying opportunities for detection. These were defined to occur at local interdistance minima between each intelligence unit and target pair. Figure 4 is an example showing when detection opportunities would be considered to have occurred. Phase II consists of identifying the terrain types intervening along the line-of-sight and determining the probability of a detection occurring. Phase III consists of having the computer flip a coin to determine if a detection occurs for any given opportunity in any given tactical model simulation. Phase IV occurs only if Phase III results in a simulated detection. If so, Phase IV predicts the result of the classification process and generates a simulated intelligence

message. The message consists of a report on the who, what, where, and when dimensions of the sighting.

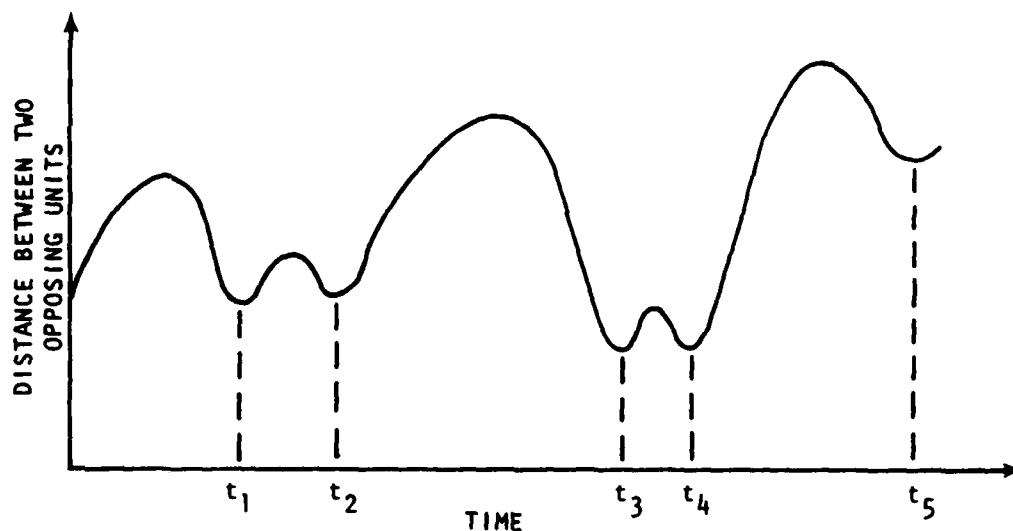


Figure 4. Example of Local Interdistance Minima at  $t_1$ ,  $t_2$ ,  $t_3$ ,  $t_4$ , and  $t_5$  That Are Input to the Detection Opportunity Model.

A mathematical algorithm was developed to determine the probability of correct detection as a function of the lengths of intervening terrain types. Values from the visibility table were used as parameters in the equation. The probability of correct detection was then used as a parameter in the confusion table which predicted probability of making a correct classification. This led to the determination of the probabilities of each possible incorrect classification. The computer then "flipped" another coin to determine the given classification for a given tactical model event. The actual equations for these models can be found in Volume II, Subsections 3 and 4 of Appendix B.\*

\*Available from the Army Research Institute.

### 2.3.5 Graphical Tactical Input/Replay

A dominating feature of the concept which is not especially a supporting model but which deserves discussion is that of the Graphical Input/Replay. When building a tactical model, the user desires a coordinated enemy tactic to result. Thus the movements and actions of each enemy unit must consider the movements and actions of all the other enemy units. But when the user can only define the movement and action of one unit at a time (in the present implementation of the concept), he needs help to insure coordination.

The pertinent control dimension is the selection of times of arrival at points along a user-determined path. The user wants to insure that the time varying distance and orientation relationships between various units achieve a tactically meaningful pattern. In order to insure this he must see a replay of the motions and actions defined up to any point in building a tactical model. The concept then is designed to be implemented in a manner that allows a compressed time graphical replay of that portion of the model defined to date. Thus, the user can interrupt his model building at any point, review via replay, and resume building. In this fashion he can easily select times of arrival for a given unit which cause it to maintain the desired relationships with other units.

### 3.0 CONCEPT IMPLEMENTATION

#### 3.1 MAN-MACHINE INTERFACE DESIGN

##### 3.1.1 Display Layout

The display screen and its contents were organized into five main components (see Figure 5).

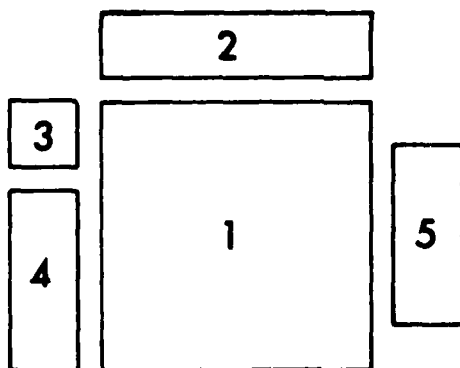


Figure 5. Display Components.

Each of these five areas plays a distinct yet interrelated part to clarify and interact with the user as the program proceeds. Although certain areas may not always be active (and in some cases not visible), the user will be highly dependent on the information displayed by each area at one time or another.

Area #1, the largest and most complex section, is active throughout the program. All variable pictorial information is displayed within this area. It represents a 20 km by 20 km geographical zone. It is within these boundaries that the terrain is defined, the order of battle is laid, and the progression of unit activity is portrayed. Basically, area #1 contains all the pictorial information that is displayed during the program flow. Path definition, movie replay of force actions, and message analysis are all pictorially presented in this section, while the contents of areas #2-5 aid in the execution of any given user task.

Area #2 is the only other pictorial section on the screen. This section is active after terrain and order of battle have been defined and completed. It then remains active throughout the remainder of the program, and consists of a time line covering a 24-hour span with hash marks every three hours. As the contents of area #1 are updated, an arrow above the time line will change position to indicate the corresponding time represented.

As the exact time reflected by area #2 is somewhat difficult to interpolate, an alphanumeric equivalent is generated in area #3. This area not only provides exact numerical values for the time portrayed in area #1 ("screen time"), but also has alphanumeric information for a "reference time" (periodically changed by user), and the "movie speed." The alphanumerics in area #3 are active during the same time span as area #2.

During the movement definition phase of the program, area #4 plays an important part in displaying alphanumeric information pertaining to mobility limitations. The arrival and departure times along specified paths are managed in this space. Other limitations with respect to crossing terrain types or engaging in fire are also handled in area #4. The area is also used to display warnings or cues to the operator to remind him of where he stands in the flow of the program.

The last area that is used on the screen, area #5, is only active during execution of the routine that enables the user to see intelligence messages. This area will display up to 10 messages (at any given time) that have been generated as a result of execution of the intelligence simulation routine. By interrogating the desired message symbol located in area #1, the corresponding alphanumeric translation will appear in area #5 and remain there until the user decides to remove it.

### 3.2 CONTROL DEVICES

The devices with which the user is able to interact with the computer/program include a trackball, light pen, function keyboard, and an alphanumeric keyboard console.

The trackball is used whenever it is necessary to designate a specific location [(x,y) coordinate] on the screen (limited by the span of area #1 in Figure 3). This would include drawing (defining) terrain features by locating endpoints of line segments comprising a polygonal contour. The trackball's movement is shadowed by a figure on the screen which repositions itself as a function of the movement of the trackball. This "figure," depending upon which portion of the program the user is in, may be a simple tracking cursor, a large circle indicating an area of indirect fire, or even a unit symbol which needs repositioning.

The lightpen plays its part when the user needs to indicate or select certain symbol(s) on the screen. This selecting process is simply a matter of pointing at the desired symbol with the lightpen. If the user wishes to define a path for a given unit (or any other function which is related to a specified symbol), the user must first point at the desired symbol, then proceed with the definition (execution) phase. Another important use of the light pen comes during execution of the routine that displays messages to the user. To specify which message is desired, the light pen is used to point at the appropriate symbol on the screen.

At any given point in the program, the user will generally have the option of choosing one of several operations. The user conveys his choice of function modes through a function keyboard. For example, if the user has chosen to execute a movie replay of the day's activities, he has four possible functions to access. He may (1) redefine starting times, (2) indicate movie speed, (3) start the replay, or (4) exit from the replay mode entirely. The function keyboard will have a lighted key for each of these possibilities, such that the activation of any given key will cause transfer of the program to

execute the particular routines associated with the desired function. The keyboard is quite an important tool throughout the program, as there is a continual need to convey to the program which routines the user would like to execute.

Finally, the device which receives the least amount of use in the software is the alphanumeric keyboard console. It is through this device that the user is able to convey alphanumeric information. This process is needed whenever the user wishes to input starting times, arrival times, reference time definitions, etc.

As an example of the interaction of the user with the devices and the flow from one device to the next, consider the act of repositioning a given unit's initial location. First the user must type in a starting time of 0000 hours on the alphanumeric keyboard. He must then indicate the desired unit by pointing at it with the light pen. Once he has selected the proper symbol, he hits "REPOSITION" on the keyboard and repositions the unit on the screen using the trackball until the desired location is reached. At this point the user indicates that the position has been located, and exits the repositioning mode by hitting "ACCEPT" on the function keyboard. Each device is used for that task at which it is most efficient, and the sequence of devices used takes into consideration a two-handed usage which minimizes human task time. Thus, the keyboard is centered with the display, the function keyboard is to the left, and the light pen and trackball are on the right.

### 3.3 INTELLIGENCE COLLECTION PLANNER OPERATION

#### 3.3.1 Overview

Assumptions that are part of the context of the Intelligence Collection Planner (ICP) use of the planning tool include:

1. The ICP knows "ground truth" past, current, and planned future order of battle and movement of friendly forces.

2. He has control over the actions of his own intelligence collection units but not friendly combat units. Thus his model of movements and actions of friendly combat units will ordinarily remain fixed throughout the use of the planning tool.
3. The ICP has no ground truth knowledge of past, current, and planned future order of battle and movement of enemy forces. He must hypothesize these.

The ICP has two tasks. One is to create a model of the tactical situation which he has in mind. The other is to devise an intelligence collection plan that is likely to be successful against a given enemy tactic or range of enemy tactics.

In developing the model of the tactical situation, the ICP uses the scenario development program (see Volume II, Appendix A-2) to:

1. Define a terrain model.
2. Represent the tactical model of the friendly combat units and their actions from information given to him.
3. Define an enemy tactical model representing suspected or likely movements and actions of enemy units.

Once the ICP has developed the various tactical models, he is ready to devise a collection plan. At the present level of software development, the collection plan consists of the movement of intelligence units. These movements are defined as the last part of the own-forces tactical model. Naturally, the ICP will define movements which are aimed at early detection of crucial events in the enemy tactical model.

Once all the models have been defined, the ICP uses the intelligence simulation program (see Volume II,<sup>\*</sup> Appendix A-3) to simulate the interaction of opposing forces and generate simulated intelligence messages. The ICP then uses the intelligence collection planner's program to replay the occurrence of the intelligence messages as they arrive in compressed time. The displayed

---

<sup>\*</sup>Available in the files of the Army Research Institute.

messages are a statistical sample of the effectiveness of the collection plan. After evaluating the results from using the plan, the ICP can revise the plan for the given enemy tactical model in the same way the plan was generated using the scenario development program. The ICP can also investigate the value of a given collection plan across a range of enemy tactics. He does this by keeping the collection plan fixed, varying the tactical model of the enemy, and evaluating the effectiveness of the plan by inspecting the sets of received messages for each model of the enemy.

### 3.3.2 Terrain Definition

Scenario development includes the definition of:

- Terrain features
- Deployment (order of battle) of the units for both enemy and friendly forces
- Movement and interaction events of opposing forces, including any battles that might occur

In addition to defining attack and defense plans, the ICP must also lay out the routes taken by his own intelligence units. These routes are the basis of any intelligence message which may be generated and displayed.

If the ICP defines terrain, then he must also define the order of battle and movement and interaction events. However, if the ICP is satisfied with a previously defined terrain model, he may begin by redefining first the order of battle and then the movement and interaction events. If he is satisfied with terrain and order of battle, he may redefine only the movement and interaction events.

The ICP defines terrain features on the 20 x 20 kilometer area represented on the display. There are six terrain types that can be defined. These are divided into two groups:

1. Closed features comprised of forests, hills, cities, and lakes.
2. Path-like features comprised of rivers and roads.

While the closed contour features are displayed by the outlines of their particular geographical limits, the path-like features have no qualities of width, and are represented by a continuous string of linear segments. Because of this, all roads are considered to be of equal width, as well as all rivers. To represent a network (i.e., river tributaries or road intersections), the user must define two separate path-like features which visually intersect. All terrain features can overlap with one another; however, it is up to the ICP to refrain from defining absurdities such as forests within forests or lakes within lakes.

The ICP chooses each terrain feature by pressing the corresponding key on the function keyboard, then drawing it on the screen with the trackball. He uses a function key to indicate the end point of each straight-line segment in the terrain feature. The software has editing provisions that enable the ICP to revise his definition of a terrain feature. After he has completed a terrain feature he proceeds to define other features until the screen contains all the features desired.

Figure 6 is an example of terrain defined by the ICP. The two five-sided orange contours near the bottom of the figure are cities. There are three hills represented in the figure by yellow contours. There is a valley between the two hills in the center of the figure. Forests are shown by green dashed contours. One forest overlaps both the central hills; the other forest overlaps the hill at the left of the figure. Roads are represented by green lines. One road goes through the central valley; the other is at the left of the left-central hill. A river represented by a red line is in the lower left part of the figure.

### 3.3.3 Definition of Order of Battle

Definition of order of battle begins after completion of terrain definition. The ICP first defines the friendly order of battle and then enemy order of battle. He can define five kinds of units by using function keys,

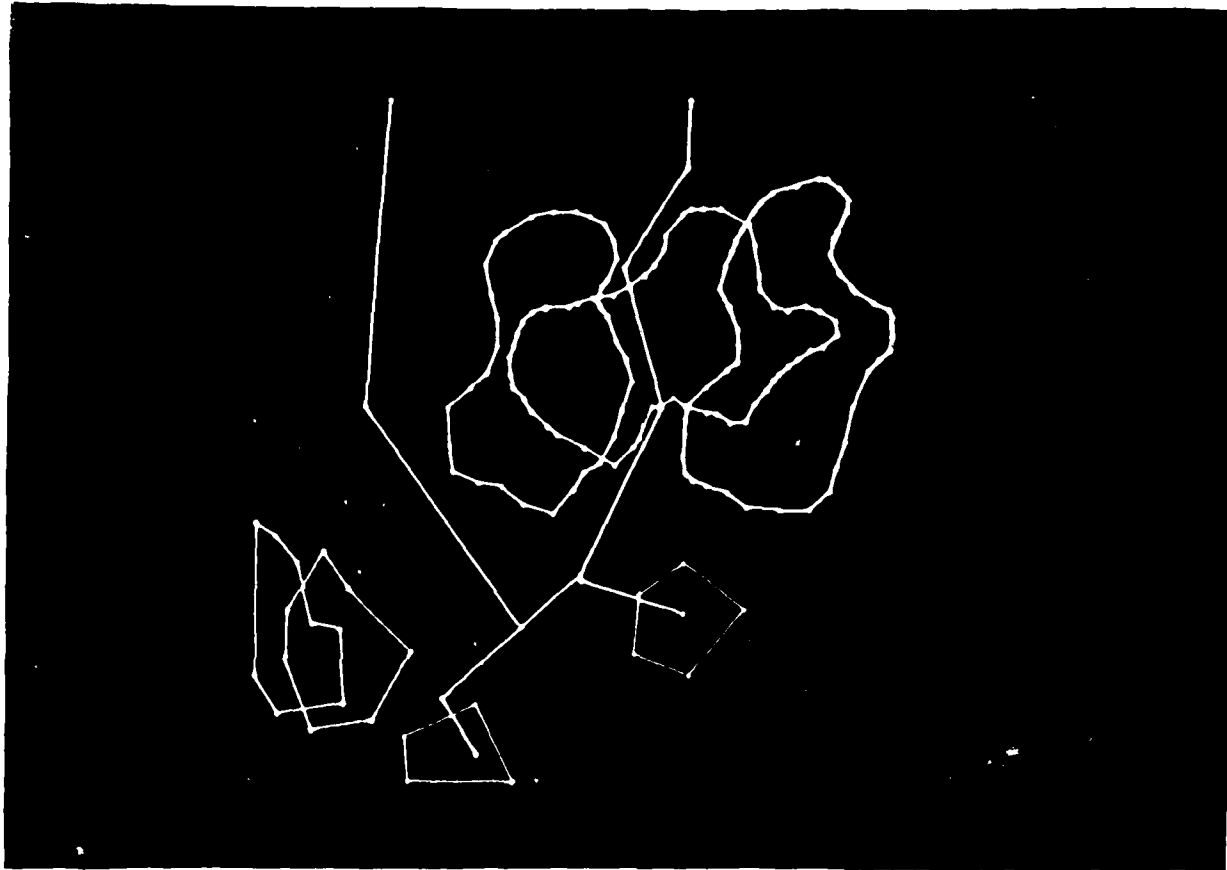


Figure 6. Example of Terrain Defined by an Intelligence Collection Planner (ICP).

namely, brigades, battalions, companies, foot patrols, and scout planes. If he specifies a brigade, battalion, or company, he must further specify the unit type, namely, infantry, armor, or artillery. After a unit has been defined, the corresponding symbol appears on the screen. The ICP then uses the trackball to position the unit where he wants it on the terrain. After friendly and enemy forces have been defined, the ICP has the opportunity to reposition any units he wishes. He does this by using editing features in the software.

Figure 7 shows the composition and locations of friendly and enemy forces at the beginning of a problem as defined by an ICP. Enemy forces are represented in red at the top of the figure and consist of two armored battalions, two armored companies, two infantry companies, and an artillery company. Friendly forces are represented in green near the bottom of the figure. They consist of two infantry battalions, an armored battalion, an armored company, an artillery company, and three foot patrol intelligence units. Each foot patrol is at the center of a circle representing the 50th percentile probability of detecting an enemy unit in open terrain.

#### 3.3.4 Definition of Movement and Interaction Events

The ICP defines all movement and interaction of opposing forces during this phase of developing the model of the tactical situation. He also implements his intelligence collection plan by specifying the routes for friendly foot patrols and scout planes. Of course, it is only possible to enter this phase of defining the complete tactical model when definitions of terrain and orders of battle have previously been entered.

3.3.4.1 Movement Events. The man-machine interface software and light pen enable the ICP to designate a specific unit on the display and a starting time for the event. Upon designation, the symbol for the unit begins to blink on the display. The ICP then specifies that he wishes to define a movement event. (His other alternative is to define a combat event.)

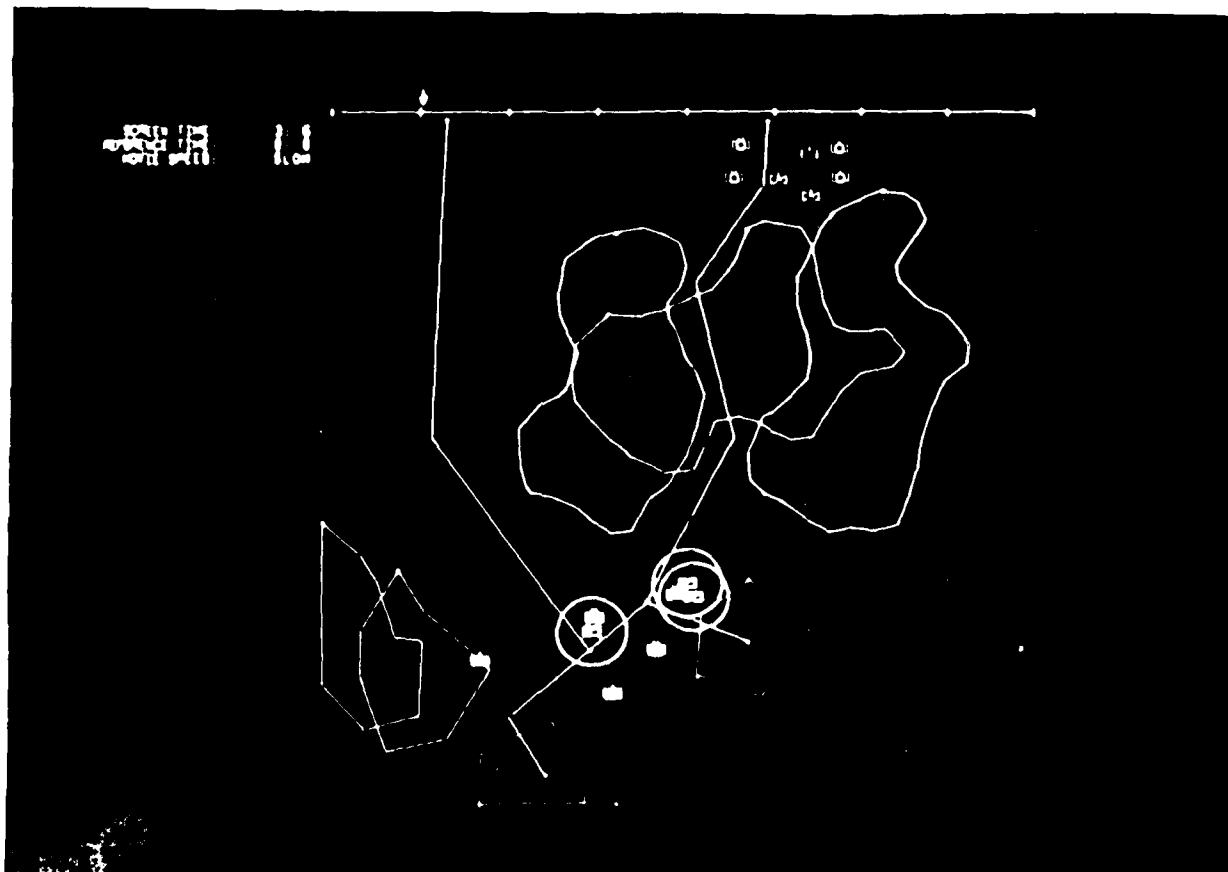


Figure 7. Composition and Location of Opposing Forces at the Beginning of the Problem.

After the ICP has specified path starting time, the computer checks to see if any path definition has been previously made for the active unit beyond the starting time specified. If so, any new path defined will destroy all previously defined path information which occurs after the starting time of the newly defined path. For example, if the ICP defines a path for unit A which takes the unit up until 3:00 o'clock, then decides to define a new path starting at 2:00 and ending at 2:30; all previous information after 2:00 will be replaced by the new path lasting till 2:30. If such an overlap occurs, the program warns the user in area #4 (see Figure 5) and prompts a decision to accept or reject the new starting time. If he rejects the new starting time, then none of the previously defined path is lost. If he accepts, the program routes itself to the path definition phase, and will erase all subsequent path information. (Note: If no overlap is originally detected, the program makes no warning, and immediately passes to the path definition phase.)

The ICP next proceeds to define the path. The path consists of a series of linear segments whose endpoints are controlled with the trackball via a rubber band line emanating either from the unit itself or a previous node on the path. After the ICP has designated the path end point, the computer checks to see if the path crosses any terrain features which the active unit may not traverse. If there is such a condition, the program tells the user and indicates where the problem exists by causing a blinking cursor to appear on the problem segment of the path. The ICP then must redefine the path to eliminate the problem.

There is a special mode called "Road Mode" for designating unit movement along a road. When Road Mode is selected, the ICP selects the desired road with the light pen and this causes a second cursor to appear on the road. This new cursor becomes the movable endpoint of the rubber band segment. The new cursor is positioned on the road as a function of the position of the original unit cursor which is still governed by the trackball. The path is now defined with respect to the road cursor. The road cursor remains until the ICP decides to leave Road Mode. This causes the original cursor to take the place of the

road cursor. Going from one road to another is accomplished by traveling down the first road to an intersection, selecting Road Mode twice on the function keyboard, indicating the second road with the light pen, and continuing on the second road.

After the ICP has completed a path by specifying its end point, the program calculates the minimum times of arrival for each node specified during the path definition with regard to terrain crossed and physical limitations of the active unit. The minimum arrival times of each node are then displayed in area #4 (Figure 5), a blinking cursor appears on the path at the arrival point of the first move or wait directive as defined earlier.

If there are 'n' segments on the path, there will be n+1 nodes with node #1 being the position of the unit at the start of the path. The table of the arrival time schedule has four headings: NODE (the associated node number), MIN (the minimum arrival time at the node), NOM (a nominal arrival time based on 80% of maximum speed), and SELECT (the arrival time as chosen by the ICP). For node #1, the SELECT column will always contain the time designated as event start time.

The table lists a nominal time of arrival only for the node currently under consideration (designated by blinking cursor), and the program waits for the ICP to define arrival times for each entry in the table. The ICP can select the minimum arrival time, the nominal arrival time, or any time later than the minimum arrival time. After he has specified an arrival time, he may also specify a waiting time at the node. Once an arrival time, or an arrival and waiting time at a particular node have been specified, the minimum and nominal arrival times for the remaining nodes are calculated and displayed. Then the next entry in the table awaits input and the blinking cursor changes position accordingly.

This set of procedures enables the ICP to define the movement of every combat and intelligence unit. An editing feature in the software enables the ICP to

redefine the movement of any unit which has previously been defined. Another editing feature allows the ICP to change the starting position of a unit from what was established during definition of the order of battle.

**3.3.4.2 Combat Events.** The ICP may specify direct fire and indirect fire for combat units. Direct fire engages the designated infantry or armor unit in one-on-one combat with another unit. (Artillery and intelligence units are considered unable to engage in direct fire.) Indirect fire engages an artillery unit in indirect combat.

When either direct fire or indirect fire is specified, a check for time overlap is made similar to that of the path definitions, but only with regard to previous firing commands. A check is also made for confirming that the active unit is capable of the specified firing type. Both checks generate diagnostic messages with the function keyboard prompting a response. The limitation on firing restricts units to only one target at a time, i.e., for any given unit, it must stop firing at its present target before beginning to fire at another.

After indicating whether direct or indirect fire is desired, the ICP must specify the beginning and ending time of fire. These choices are also checked for logic errors (there cannot be an end fire without an associated begin fire, etc.) and the ICP is warned accordingly. If all checks prove no error, the targets are then defined. For direct fire, the light pen is activated, and a selection of the target unit is made. For indirect fire, a rubber band line attached to a circle indicating barrage area may be moved until the desired area of attack is located by the circle.

**3.3.4.3 Replay.** The ICP can use a movie replay capability after he has defined any set of movements and/or combat events. Replay gives a dynamic representation of events on the screen (area #1) beginning from a specified point in time. It enables the ICP to determine if the defined events coordinate properly and truly represent the tactical situation he has in mind. Movie

speed refers to the relative rate at which a given time span of the 24-hour period covering the simulated scenario is presented to the viewer. There are four movie replay speeds to choose from, namely, very slow, slow, fast and very fast.

Screen information displayed to aid the ICP includes a time thermometer in area #1 above the area containing the terrain, and alphanumeric information regarding movie speed, "screen time," and "reference time." The distinction between screen time and reference time is as follows: Screen time is the time which is being portrayed in area #1 of Figure 5. For example, screen time is 3:12 if and only if the contents of area #1 represent the activity and locations of the units at 3:12 as defined by the ICP. Reference time, however, is merely a convenience tool. It is simply a time that the ICP chooses, and is free to change whenever necessary. This enables the ICP to repeatedly view a replay starting from the same reference point in time (hence the name), without needing to explicitly type in the same starting time for each repetition. The value of reference time is in no way affected by the contents of area #1. When the ICP inputs a new reference time, the screen time and contents are set to the new reference time.

The ICP has two replay options, namely, "Run Screen" or "Run Reference." Run Screen causes the replay to begin at a starting point designated by the screen time. In other words, action will continue from the moment that is portrayed on the screen in area #1. Run Reference, however, causes the screen contents to jump back to the time listed as the current reference time before beginning movie replay. Therefore, if the screen time is 6:02 and the reference time is 4:51, Run Reference causes a movie replay beginning from 4:51. There is also a "Pause/Resume" control. The first use of this control will momentarily "freeze" the screen for closer scrutiny. The second use causes the movie replay to continue.

Figure 8 shows a replay of the intelligence collection plan which has been stopped at 3:27 hours into the problem. The ICP has sent one patrol

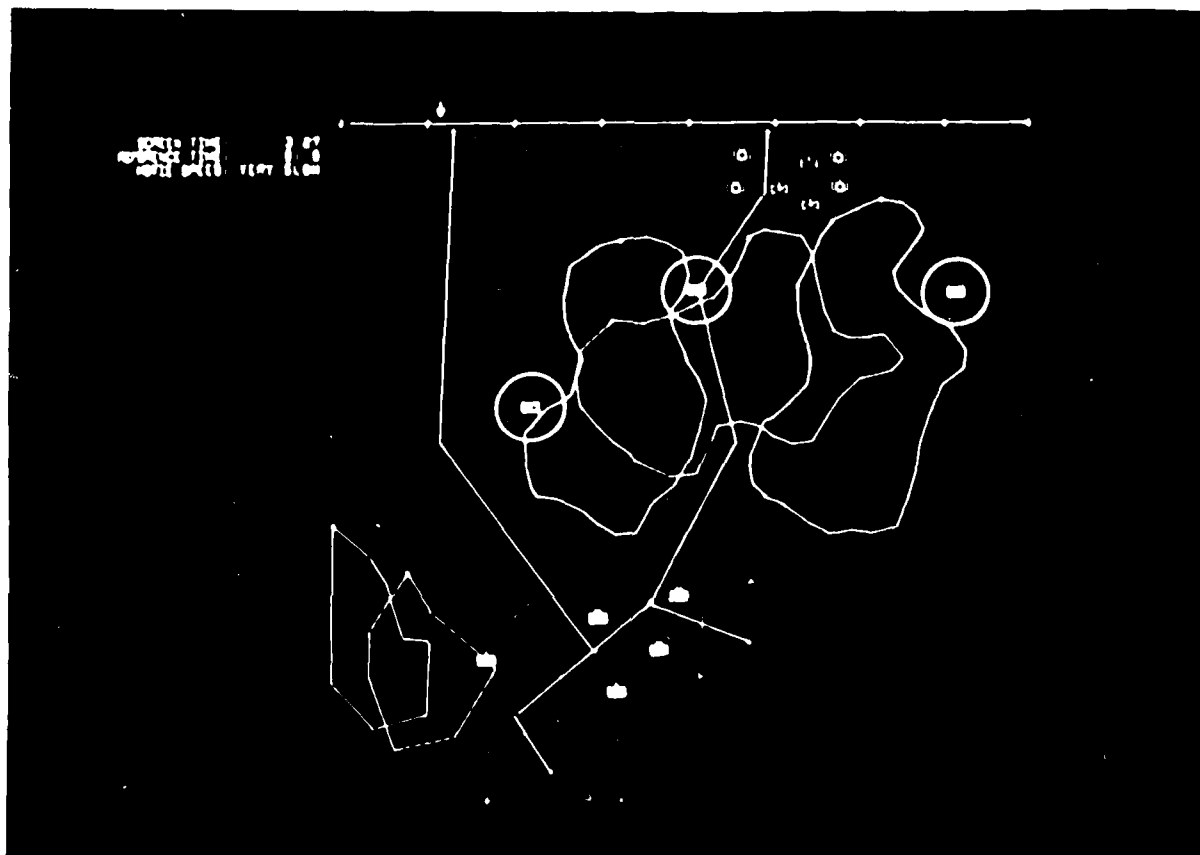


Figure 8. Positions of Intelligence Collection  
Foot Patrols 3:27 Hours into the Problem.

up the central road, one up the right side of the right-central hill, and one up the left side of the left central hill.

Figure 9 shows the positions of enemy forces 5:49 hours into the problem as hypothesized by the ICP. Two armored companies have come around the right side of the right-central hill and two armored battalions have moved around the left side of the left-central hill. The two infantry companies are on either side of the central road but they are just inside the forest. The artillery company has opened indirect fire on the area just outside the central town. This is indicated by the red circle and line between the red circle and the artillery company.

Figure 10 shows the positions of both forces at 3:43 hours. Note that the rightmost friendly patrol is in position to detect the two armor companies, and the central patrol has recently passed the two infantry companies.

### 3.3.5 Generation and Use of Intelligence Messages

The terrain, friendly and enemy order of battle, movement events and combat events constitute the tactical model of the situation envisioned by the ICP. The complete tactical model is the input to the message generation (simulation) routine in the computer program. The output of this routine is presented to the ICP as messages generated by friendly forces. As explained in Section 2.3.3., the detection model uses probabilities of detection and non-detection when enemy units come within range of a friendly unit to determine if a detection actually occurs. As in real life, if a detection occurs, there are probabilities of correct and incorrect classification. If incorrect classification occurs, the classification model determines the unit type and size for which the detected unit was mistaken.

All intelligence messages have an associated time when the message was reported. The intelligence sighting, i.e., detection, may occur earlier in time. Messages are delayed from time of sighting with delay time uniformly distributed between 0 and 120 minutes. As the day's activities unfold to

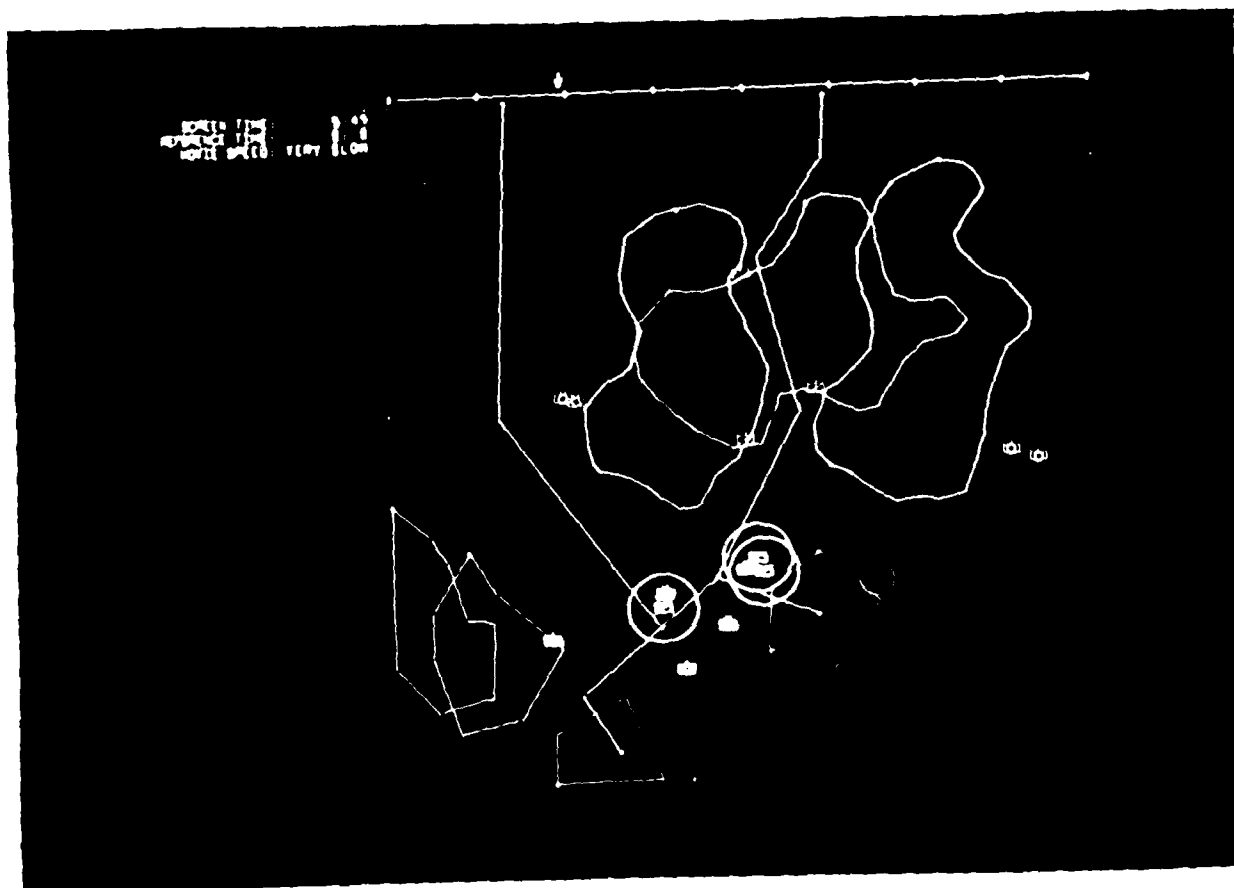


Figure 9. Positions of Enemy Forces 5:49 Hours into the Problem.

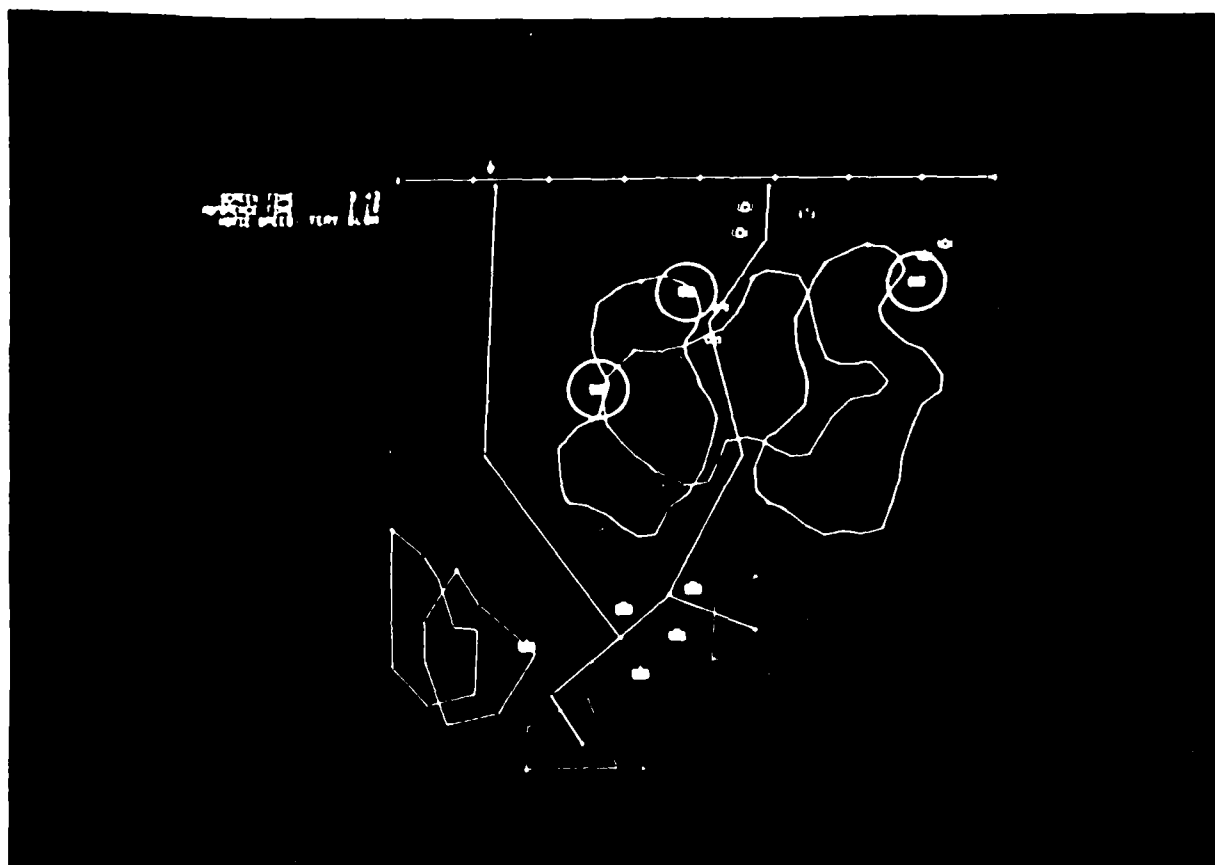


Figure 10. Position of Opposing Forces 3:43 Hours  
into the Problem.

the ICP, symbols appear on the display where each sighting was made. On the first replay of the day's events, a diamond blinks at the reported sighting location until the ICP interrogates the report with the light pen. Then the diamond will become a symbol signifying the reported type and size of that unit. In the right of the screen (area #5) will appear alphanumeric code with the information about reported type, size, and action of the unit, location coordinates, and time of sighting. The location (in kilometers from lower left corner) and time of sighting represent perfect information. The reported type, size, and action of the unit may have errors. For example, an armored company stopped will not be reported as a reconnaissance plane but may be reported as an infantry company in motion.

Since a maximum of ten messages may be displayed at any one time, the option to delete a particular message display is given. This is accomplished by using the light pen to indicate the alphanumeric message to be deleted. The alphanumeric code will disappear and the combat symbol will revert to a diamond. Deletion from the display of a message does not destroy this information. The ICP may interrogate this same message again as many times as desired. To aid the subject intelligence analyst in identifying the alphanumeric message with the combat symbol displayed, the program numbers the messages and places this number next to the corresponding combat symbol.

Upon completion of an initial replay, the ICP may want to see other replays. When a replay is begun again, all symbols representing reported enemy units revert to diamonds. Then, as the replay proceeds, the diamonds turn to crosses as screen time reaches time of sighting. In this fashion, the ICP can see the reported combat units as replay time progresses plus the locations where the remaining reports will occur.

The ICP evaluates his collection plan in terms of the set of displayed messages. If the messages provide timely detection of hypothesized enemy Tactic A, his plan is good for that tactic. If not, then the ICP will revise

his plan until it provides a satisfactory set of messages. Once he has a satisfactory plan for Tactic A, the ICP will probably want to investigate the adequacy of the plan for other potential enemy tactics. Thus he may ultimately change a plan which was excellent against enemy Tactic A but poor against Tactic B to one that is good for A and fair for B.

Figure 11 shows the results of a replay at 3:53 hours. The hypothesized units in the enemy order of battle are shown in red at the bottom of the screen. A total of ten sightings were made during the problem. The unit types and sizes are shown in orange symbols for three of the reports. The alphanumerics in the messages shown at the right of the picture correspond to these three symbols. The orange diamond just below the symbol for an Infantry company on the central road and the five orange diamonds below the middle of the display represent sightings that took place after 3:53 hours. The orange cross to the right of the right-most patrol represents a sighting made before the screen time of 3:53 hours. The combination cross and diamond near the top of the picture represents a sighting made before 3:53 and tells the viewer that another sighting of the same unit will also occur after 3:53 hours.

#### 3.4 INTELLIGENCE ANALYST OPERATION

Assumptions that are part of the context of Intelligence Analyst (IA) use of the analysis tool include:

1. All information used by the IA is past information. He knows ground truth about friendly order of battle, movements of friendly combat and intelligence units, and combat events between friendly and enemy forces. These are inputs that he can make to the overall tactical model.
2. The IA has a set of actually received intelligence messages about enemy forces.
3. The IA has no ground truth knowledge of enemy order of battle and tactics. He must create a tactical model(s) of his hypothesis(es) about the enemy.



The IA has two tasks. One is to create a complete model of the tactical situation as he envisions it. The second is to decide which enemy tactic is most consistent with the intelligence message he has received.

In developing the model of the tactical situation, the IA uses the scenario development program (see Volume II, Appendix A-2) to:

1. Define a terrain model.
2. Represent by means of a tactical model his ground truth knowledge of own-forces order of battle and movement events for both combat and intelligence units.
3. Define a tactical model of the enemy representing suspected or likely past movements and actions of enemy forces.

These tasks are performed using the same software and in the same manner as described in Section 3.3 for the Intelligence Collection Planner. Once the complete tactical model is defined, the IA inputs this to the message generation (simulation) routine in the computer just as the ICP does. He also views, interrogates, and replays the intelligence messages as the ICP does.

The major differences between ICP and IA operation are:

1. The IA is dealing with fixed (past) actions of the intelligence units. The ICP varies the actions of the intelligence units in order to develop the best plan.
2. Both IA and ICP develop models of hypothetical enemy tactics. However, the ICP investigates different enemy tactics in order to devise the best intelligence collection plan. The IA investigates different enemy tactics in order to determine which one is most likely to have produced the set of actually received intelligence messages.

#### 4.0 EVALUATION AND RECOMMENDATIONS

Use of the software to debug various portions, provide presentation examples for the Project Monitor, and develop a 35mm slide presentation provided ample opportunity for qualitative evaluation. The three major results of the evaluation are:

1. Much refinement needs to be made to the coefficients in the various models (e.g., movement, intelligence detection, and intelligence classification) to improve the validity and consistency of the simulations.
2. The rule for identifying detection opportunities was too simplistic. (The rule was that a detection opportunity occurred when a pair of opposing units reached an interdistance minimum. See Section 2.3.4.) For example, if the distance between two opposing units was as shown in Figure 12, an opportunity to detect would not occur until  $t_1$ , even though the distance between the units was very small for a considerable period prior to  $t_1$ .
3. There is excessive tedium involved in defining tactical models by addressing the movement and action of each unit independently.

The first recommendation arising from the evaluation is to refine the model coefficients. The second recommendation is to increase the sophistication of the detection logic so that detections occur at realistic times and under realistic circumstances.

The third recommendation focuses on the tedium problem. Since most military tactics consist of time-phased coordination of a limited number of well-established elements, what is required is the development of software functions to provide those elements. Such tactical "rules-of-thumb" can be developed at an increasing level of detail and comprehension.

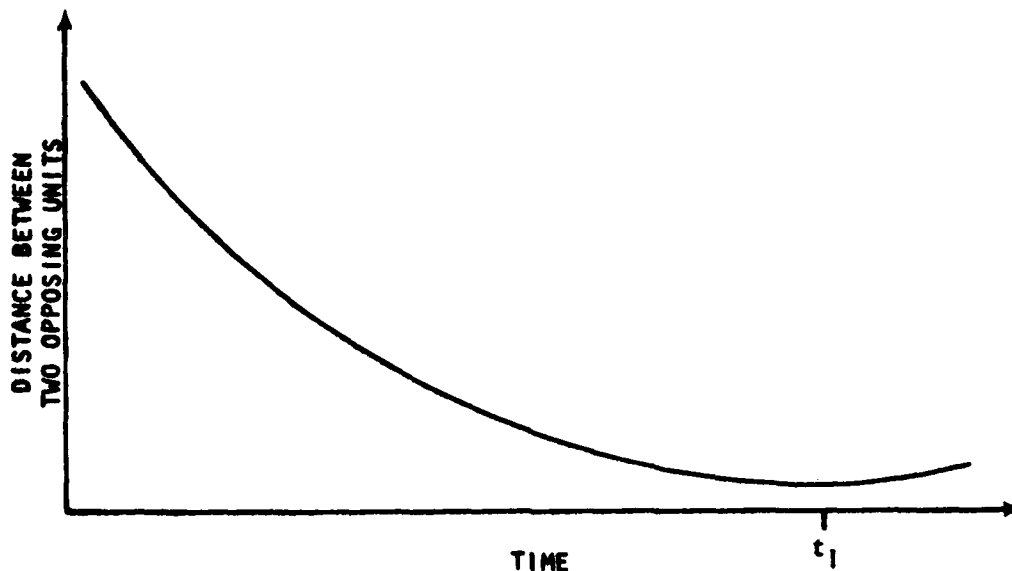


Figure 12. Example of a Situation where the Existing Detection Opportunity Rule Produces a First Opportunity to Detect That Is Unrealistically Late.

What has been developed to date, the software for defining the movement and actions of each individual unit, comprises the "primitives" for such "rules-of-thumb." These can be combined to develop the next level of comprehension, those tactical actions involving a few units, or a simple coordinated action. An example is a battalion moving in column file down a road. Instead of the software user having to define the path and rate of motion for each individual company, he should be able to address the software at the battalion level, and then the software would translate this into the details for each of the involved companies.

As each new set of tactical "rules-of-thumb" uses the previous set as its library of primitive commands, more and more comprehension can be built into a single operator action. Eventually, it is desired to allow the operator to deal at the battalion or brigade level only and have to go to the company level only when a unique tactical refinement is required. Thus, it is recommended that a multi-year program be instituted for developing and evaluating tactical "rules-of-thumb."

There are two other recommendations. These do not arise from evaluation of the existing ICP and IA tools but rather from the experience gained from developing the tools and the knowledge of what else could be done using interactive graphics. The first such recommendation is to reorganize the existing software so that the value of the tools can be tested by experiment. (This can be done at very slight cost.)

The second recommendation is to develop and evaluate an interactive computer concept to permit the graphical analysis of a large and varied intelligence data base and relate the results of the analysis to the existing battlefield tactical dynamics. This differs from the work done on the existing contract in the following way: The current work focused on enabling the user to input and evaluate tactical hypotheses about enemy actions. The recommended work would focus on finding the best ways of accessing and presenting existing intelligence data. It would involve the following:

1. Designing a candidate intelligence data base that is sufficiently comprehensive, readily accessible, and could serve as the data source for interactive graphics software.
2. Designing and implementing a flexible data base interrogation software routine that can:
  - (a) access the file according to the operator-prescribed dimensions of the resident data
  - (b) perform basic correlation and logical operations
  - (c) structure the output in a form usable by candidate man-machine interface (MMI) designs.
3. Designing and implementing several candidate interactive graphics MMI designs that would permit the IA to efficiently access the data base and obtain the output in both tactical space (e.g., map, geographic) and synthetic (e.g., graphs, bar charts, patterns) formats.
4. Performing a preliminary engineering/psychological evaluation of the system and its alternative MMI designs in the context of a tactically dynamic battlefield scenario.